Software Requirements Specification

Software Engineering & Project

Team: PG-29
Phan Huy Nguyen
Sean Hennessy
Kin Leong Lee
Pavitterjeet Sidhu
Benjamin Winding
Xiaoshan Chen

03/08/2017

Contents

1 Introduction

1.1 Purpose

This report is to describe the software requirements specification of the lunar rover project.

1.2 Document Conventions

The document conventions in this project include the following main points....

1.3 Intended Audience and Reading Suggestions

The intended audience of this document are the Client, this project team and project supervisor.

1.4 Project Scope

The scope of this project is to demonstrate a prototype for a lunar robot, which is capable of performing an automated survey of a extraterestrial landscape.

This robot is to be constructed using the EV3 Lego Mindstorms robot provided by the client. It is to be controlled via a remote location, but is required to automatically make decisions based on the environment around it.

1.5 References

1. Client Specifications,
2. Project Specifications,
3. Ev3 kit

2 Overall Description

2.1 Product Perspective

The product described in this SRS is new self-contained project. It relies on the existing LEGO robot that is built by all LEGO bricks. Previously, robot cannot be controlled by human being with software no matter whether manually or automatically. In addition, this SRS defines the component of the robot system and the following diagram is identifying the functionalities and interfaces of this system.

2.2 Product Features

The main features of this robot include

- Enables the robot to map the survey area;
- Enables the robot to be controlled manually and automatically;
- Enables the robot to avoid obstacles;

- Enables the robot to detect the no-go zone;
- Enables the robot to start from the landing point and survey;
- Enables the robot to return the start point.

2.3 User Classes and Characteristics

The users of the robot include three types: students, teacher and people who only have basic knowledge of GUI control. The last type usually contains student and teacher.

Students are those who enrolled in the university of Adelaide, especially the school of computer science. There are constraints for students: they are not allowed to access the robot system, and they can only control the robot after the group members enter the password for them.

Teachers are permitted to get the password of the robot system and should not tell students the password.

People who has basic knowledge of GUI control can also control the robot and they cannot get the password as well except for the client of this project. The client owns all privileges of this robot as long as the robot is not damaged deliberately.

2.4 Operating Environment

The software can be run in Window7 or above, Mac OS 10 or above, Linux and Ubuntu 16+ as long as GUI is complied using JDK (version 1.7 only). The information of hardware includes:

- Window: 1 gigahertz (GHz) or faster 32-bit (x86) or 64-bit (x64) processor*; 1 gigabyte (GB) RAM (32-bit) or 2 GB RAM (64-bit); 16 GB available hard disk space (32-bit) or 20 GB (64-bit); DirectX 9 graphics device with WDDM 1.0 or higher driver.
- Mac: An Intel Core 2 Duo, Core i3, Core i5, Core i7, or Xeon processor; Mac OS X v10.6.6 or later to install via the Mac App Store (v10.6.8 recommended); 7 GB of available disk space; 2 GB of RAM.

2.5 Design and Implementation Constraints

The main constraints of design and implementation include:

- This project is restricted using tools and the group use particular tool to do particular thing. For example, firstly, when identify the functionalities of the robot, IntelliJ is used to design and write code. Secondly, Github is used for version control and the members of group share files each other by commit. Thirdly, slack is for communication to exchange ideas or discuss some issues. Fourthly, the tool of testing functionalities of robot is ant. Also, Python is considered as writing the test script. Lastly, Latex is used for documentation since it is simple to implement professional documentation;

- The operating environment should meet the minimum requirements of hardware and software that mentioned in section 2.4;
- The version of LeJos software should be 0.9.0;
- The version of Java should be 7. It does not yet support version 8;
- Since the controller is connected with Wi-Fi and only allows the user to access, the encryption is necessary.
- The software only supports Lego mind storm EV3;
- The external code should not exceed 10 %.

2.6 User Documentation

At the end of the project, a user manual would be available to users on a handbook. The handbook will mainly describe the different parts of GUI, including the getting started window, the paths and targets browser, the offline and online browser and so on, assisting users to control the robot better.

2.7 Assumptions and Dependencies

It is assumed that the password is entered correctly and the robot establishes connection successfully with appropriate software and hardware. Also, the robot is ready to be control manually or automatically.

Assumptions and dependencies include:

- Any survey area can be landing zones;
- The map has clear colours of track/trails;
- The map is flat surface;
- Thickness of the line is 2cm;
- Trail is reasonably smooth;
- Survey area is rectangular;
- Landing zone does not contain no-go zone and dangerous area.

3 User Requirements

3.1 The Map

3.1.1 The robot shall enter the survey area and produce a survey map

Description Upon landing, the robot shall enter the survey area and produce a survey map based on the track followed by the vehicle.

Rationale This map will be used to analyse the survey area. This map will also be used to monitor and provide instruction to the vehicle.

Acceptance criteria This requirement can be verified by doing a survey of an area by using the software that we are going to develop and the survey map will be automatically produced.

3.1.2 The map shall be constructed in real time

Description After landing the robot on the predefined landing zone, the map shall be constructed in real time as long as the robot starts the survey.

Rationale This feature will allow the remote operator to monitor and instruct the robot in real time.

Acceptance criteria This requirement can be verified by physically observing the robot and compairing the changes on the map.

3.1.3 The current location of the vehicle shall be visible on map

Description The current location of the vehicle shall be clearly visible on the map in real time. No matter where the robot goes in the map, the operator shall be able to track the current locaton.

Rationale This feature will allow the remote operator to track the current location of the robot.

Acceptance criteria This requirement can be verified by moving the robot into different directions and physically comparing the actual position of the robot between the physical map and the map on the GUI .

3.1.4 The map shall be able to be stored in "XML" format

Description The survey map which shall be created during the process of survey, shall be able to be stored in "XML" format in the system.

Rationale This feature will allow the user to store, process, share etc the survey map.

Acceptance criteria This requirement can be verified by storing any sample survey map in "XML" format and then veiwing it in "XML" supported plateform only.

3.1.5 The software shall allow to load existing "XML" map file

Description The user shall be able to load any existing partial or fully completed suvey map file in "XML" format into the software.

Rationale This feature will allow the user to test the robot before sending it to actual survey area and will also be handy in case of system failure.

Acceptance criteria This requirement can be verified by loading an exiting "XML" map file in the software and by allowing the robot to suvey this map.

3.1.6 The map shall allow to zoom in on particular area

Description The user shall be able to zoom in on particular area in the map by using the GUI of the software.

Rationale This feature can be very useful if the user wants to focus on a particular area on the map and wants to have a closer and detailed look.

Acceptance criteria This requirement can be verified by client by zooming in on a particular area of a sample map.

3.1.7 The map shall allow to designate NGZ any time

Description The remote operate shall be able to designate NGZ any time on the map by using the GUI of the software.

Rationale This feature can be very useful in avoid any potential dangerous areas of the map detected by the remote operator.

Acceptance criteria This requirement can be verified by drawing a few NGZ on a sample map using GUI.

3.2 Sensors

3.2.1 The robot shall not go into craters

Description The robot shall detect and avoid going into carters during the survey and try to find any alternate route to destination.

Rationale This feature will protect the robot from going into area where it is impossible for robot to recover without assistance.

Acceptance criteria This requirement can be verified by trying to send the robot into craters by setting the destination on other side of the carter.

3.2.2 The robot shall not collide against an external object

Description The robot shall detect an external object and avoid colliding with it .

Rationale This feature will protect the expensive vehicle from harm and also preserve the integrity of the survey site..

Acceptance criteria This requirement can be verified by instructing the robot to move towards an external object.

3.2.3 The robot shall not go into any NGZ on map

Description The robot shall detect and avoid going into any NGZ available on the survey map.

Rationale NGZ is considered as potentially dangerous area of the map in which robot shall not go. By doing so, the robot can be protected from any harm.

Acceptance criteria This requirement can be verified by trying to send the robot into NGZ by setting the destination of the robot on other side of the NGZ.

3.2.4 The robot shall be able to detect a track on a given map

Description The robot shall be able to detect and follow any given track on the survey map.

Rationale This will allow the operator to survey the specific area of the map.

Acceptance criteria This requirement can be verified by loading an existing survey map with atleast one track into software and robot will detect and follow that track.

3.3 Operations

3.3.1 The robot shall return to landing site

Description The robot shall remember its landing site and shall return to it when the survey is finished.

Rationale This feature will allow the operator to bring the robot back, once the survey is finished.

Acceptance criteria This requirement can be verified by instructing the robot to return to the landing site at any time during the survey.

3.3.2 The software shall provide manual control of the robot as well

Description The operate shall be able to take manual control of the robot at any time.

Rationale This feature can be useful in manually adjusting the position of the robot at any time.

Acceptance criteria This requirement can be verified controlling the robot manully at any time using the provided GUI.

3.3.3 The operater shall be able to stop the robot at any time

Description The remote operator shall be able to stop the robot at any time using the stop button on the GUI.

Rationale This feature can be useful in protecting the robot or if the operator wants to abort the mission, he/she can just stop the robot and instruct it to return to the landing site.

Acceptance criteria This requirement can be verified by pressing the stop button on GUI and in response to this, the robot will stop immediately.

3.3.4 The robot shall be able to move immediately to a given point

Description The remote operator shall be able to place survey point on the map and the robot shall move immediately towards that point.

Rationale This feature will allow the operator to survey any area on the map by just setting the survey point on the map.

Acceptance criteria This feature can be verified by setting a survey point on the map using the GUI and the robot will move towards it.

4 System Features

5 External Interface Requirements

5.1 User Interfaces

5.2 Hardware Interfaces

5.3 Software Interfaces

5.4 Communications Interfaces

6 Other Non-Functional Requirements

6.1 Performance Requirements

Map Accuracy
Description: The visual representation of the map shall be as accurate as possible. The various objects and hazards that the robot encounters must be recognized and drawn as soon as the robot encounters them.
Rationale: The client must be able to use the map for other purposes once the crash site has been found and thereofre is relying on the map to be an accurate representation of the environment. In addition, the robot must be able to use the map to determine where past obstacles were to allow for smooth navigation and to avoid unnecessary travel time.

Speed
Description: The surveying of the land and discovery of the goal shall be completed in reasonable time. For this prototype, this shall be interpretted as no more than 25 minutes to complete the journey form landing to returning to the landing zone after completeing the goal. On manual control this shall be determined by the operator of the robot but will not be allowed to exceed a

specified limit for safety purposes.
Rationale: The robot will have a limited power supply and must be able to complete its mission before that power runs out.

6.2 Safety Requirements

Significant Impact
Description: While on autopilot, the robot shall not exceed $0.3m/s$. When detecting an obstacle, the robot shall stop within 0.15 metres before collision to avoid significant impact and to allow for turning room. During prototype testing, a collison will be deemed significant impact if the robot manages to physically move an obstacle. The manual controller shall default to no more than $0.3m/s$ however the pilot may have additional speed options available to them.
Rationale: The safety of any persons on the landing site is paramount and a collison of the actual robot to any persons may lead to severe injury or death. The prototype is also representing a robot that will be expensive to repair and significant damage to the robot may lead to irrepairable damage and failure of the mission.

No Go Zones
Description: No more than half the robot may enter the NGZ at any time.
Rationale: An NGZ may represent an area where personnel are active or may indicate an area of known hazardous materials or obstacles otherwise not detectable by the robot. For the safety of the personnel and the robot, the prototype must be able to demonstrate the ability to maneuver its way out of an NGZ without risking injury to personnel or getting stuck in an hazardous area.

6.3 Security Requirements

Unauthorized Access

Description: The GUI shall have a login section where a username and password is presented and verified before the controller can be used.
Rationale: Unauthorized use of the robot may lead to damage or loss if targeted by malicious competition or an individual who has not been verified by the client.

6.4 Software Quality Attributes

Ease of use
Description: The GUI shall have intuitive controls and clearly marked buttons. The size of the contoller and map shall be set to a size that is sufficient for comfort and legibility. The visual style of the GUI will conform to the standards of similar controller layouts commonly found in maps and remote controllers.
Rationale: The Client will have limited time to spend learning controls and

must be able to intuitively navigate the GUI without much training. This should include importing and exporting map data.

7 Other Requirements

8 Appendix A: Glossary

NGZ: No-Go Zone

9 Appendix B: Analysis Models

10 Appendix C: Issues List